

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-98719

(43) 公開日 平成10年(1998) 4月14日

(51) Int.Cl. ⁶	識別記号	F I	
H 0 4 N	7/24	H 0 4 N	7/13 Z
G 0 6 T	11/00	H 0 3 M	7/30 Z
H 0 3 M	7/30	H 0 4 N	1/41 B
H 0 4 N	1/41	G 0 6 F	15/72 A
	5/92	H 0 4 N	5/92 H
審査請求 未請求 請求項の数 5 F D (全 17 頁)			

(21) 出願番号 特願平8-271909

(22) 出願日 平成8年(1996) 9月20日

(71) 出願人 395015319

株式会社ソニー・コンピュータエンタテインメント

東京都港区赤坂7-1-1

(72) 発明者 鈴置 雅一

東京都港区赤坂8丁目1番22号 株式会社
ソニー・コンピュータエンタテインメント
内

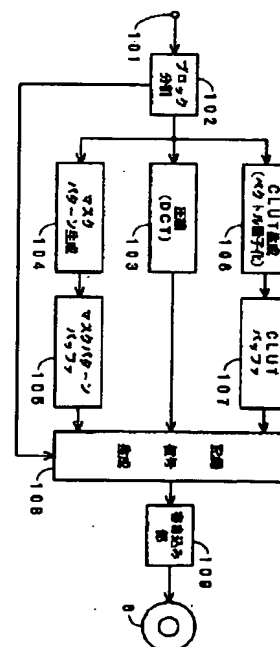
(74) 代理人 弁理士 佐藤 正美

(54) 【発明の名称】 画像データ伝送方法および画像処理装置並びに記録媒体

(57) 【要約】

【課題】 高画質の画像を非可逆圧縮して伝送しても、伸長デコード後の画像として、透明画素は正しく透明画素として復元できる。

【解決手段】 画素値として透明を表現する値が割り当てられた画像データであって、予め定められた所定の大きさの画面領域分の画像データ単位で、データ圧縮して画像データを伝送する。画像データ単位の元の画像データのそれぞれの画素について、それが透明であるか不透明であるかを識別する付加データとしてのマスクパターンを、データ圧縮された画像データ単位との対応関係をもって伝送する。データ圧縮された画像データを伸長したときに、マスクパターンにより透明画素とされている画素は、伸長画像データに関係なく、強制的に透明画素とする。



BEST AVAILABLE COPY

【特許請求の範囲】

【請求項1】画素値として透明を表現する値が割り当てられた画像データであって、予め定められた所定の大きさの画面領域分の画像データ単位で、データ圧縮して画像データを伝送すると共に、

前記画像データ単位の元の画像データのそれぞれの画素について、それが透明であるか不透明であるかを識別する付加データを、前記データ圧縮された画像データ単位との対応関係をもって伝送することを特徴とする画像データ伝送方法。

【請求項2】前記データ圧縮された画像データを伸長したときに、前記付加データにより透明画素とされている画素は、伸長画像データに関係なく、強制的に透明画素とすることを特徴とする請求項1に記載の画像データの伝送方法。

【請求項3】画素値として透明を表現する値が割り当てられた画像データであって、予め定められた所定の大きさの画面領域分の画像データ単位で、データ圧縮された画像データと、前記画像データ単位の元の画像データのそれぞれの画素について、それが透明であるか不透明であるかを識別する付加データとが対とされた入力画像データの処理装置であって、

前記データ圧縮された単位画像データごとにデータ伸長する伸長デコード手段と、

前記伸長デコード手段からのデータ伸長された単位画像データごとに、前記付加データにより透明画素とされている画素は、強制的に透明画素とする画像補正手段と、を備えることを特徴とする画像処理装置。

【請求項4】画素値として透明を表現するための値が割り当てられ画像データにより、透明を含む描画を行う画像処理装置において、

前記透明の画素に隣り合う不透明の画素は、半透明にして描画を行うことを特徴とする画像処理装置。

【請求項5】予め定められた所定の大きさの画面領域分の画像データ単位で、データ圧縮された画像データが記録されると共に、

前記画像データ単位の元の画像データのそれぞれの画素について、それが透明であるか不透明であるかを識別する付加データが、前記データ圧縮された画像データ単位との対応関係を持って記録された記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明は、例えば、ビデオゲーム機やパーソナルコンピュータのように、高い精度は求められないものの、限られたハードウェア資源の中で、ユーザの入力や演算結果に応じてリアルタイムに3次元物体を表示することが求められるシステムにおける画像データ伝送方法および画像処理装置、さらには、画像伝送用の記録媒体に適用して好適なものに関する。

【0002】

【従来の技術】家庭用TVゲーム機やパーソナルコンピュータあるいはグラフィックコンピュータなどにおいて、TV受像機やモニタ受像機などに出力して表示する画像データを生成するための画像処理装置は、バスを通じて互いに接続される、汎用メモリ、CPUやその他の演算LSIが組み合わされて構成され、描画用の画像データの流れとして、CPUとフレームバッファの間に専用の描画装置を設けることにより、高速処理を可能にしている。

10 【0003】すなわち、上記画像処理装置においては、CPU側では、画像を生成する際に、直接、表示画面に対応する表示メモリとしてのフレームバッファにアクセスするのではなく、座標変換やクリッピング、光源計算等のジオメトリ処理を行ない、3角形・4角形などの基本的な単位図形（ポリゴン）の組合せとして3次元モデルを定義して3次元画像を描画するための描画命令を作成する。そして、CPUは、その描画命令を外部バスを介して描画装置に送る。

20 【0004】描画命令には、描画しようとするポリゴンの形、位置、向き、色、模様などの情報が含まれる。ポリゴンの形、位置、向きは、その頂点の座標で決まる。

【0005】上記画像生成装置において、例えば、3次元のオブジェクトを表示する場合は、当該オブジェクトを複数のポリゴンに分解して、各ポリゴンに対応する描画命令をCPUが生成し、その生成した描画命令をバスを通じて描画装置に転送する。描画装置は、当該描画命令を実行して、フレームバッファに表示データを書き込み、目的の3次元オブジェクトを表示する。

30 【0006】この際に、オブジェクトをより実際に近く表現するために、所定の画像パターンを用意してその画像パターンによりポリゴン内部を修飾するテクスチャマッピングやミップマッピングと呼ばれる手法が採用されている。

【0007】さらに、色変換データを記録した色変換テーブル（以下CLUT（Color Look Up Table）と称する）を介して画像の色データを変換することにより、表示色を変化させる手法も広く知られている。

【0008】

40 【発明が解決しようとする課題】ところで、家庭用TVゲーム機や、パーソナルコンピュータを構成する演算LSIは、コストの上昇なしに、その動作周波数の高速化、回路規模の縮小などの性能向上が計られているが、コスト上昇なしで使用できる汎用メモリの容量はさして増加していない。このため、家庭用TVゲーム機や、パーソナルコンピュータでは、メモリ容量がいわゆるボトルネックとなっている。

50 【0009】特に、前もって高性能のワークステーションで作成された高品質の画像（プリレンダリングパターン）をテクスチャパターンとしてテクスチャマッピング

をする場合、一連の動画の切片を構成するパターンをすべてメモリに保持しておく必要があるが、テクスチャパターンの解像度が上がれば、それに伴いメモリ容量が圧迫されることになる。

【0010】そのため、テクスチャパターンをデータ圧縮してメモリに保持するようにすると共に、圧縮されたテクスチャパターンのデータを使用の都度、メモリから読み出し、専用の画像伸長装置（圧縮を解凍する装置）を用いて、伸長デコードして使用する方法がとられている。

【0011】ところで、テクスチャパターンは、例えば、縦×横＝64画素×64画素の大きさの画面領域分の画像データで構成される。しかし、テクスチャパターンは、矩形であるとは限らないので、前記の矩形の画面領域分の画像データの内に、描画しない画素、つまり透明の画素を設定して、必要なテクスチャパターンを表すようにしなければならない。

【0012】そこで、描画装置において、前記の矩形領域分のテクスチャのデータについて描画しない画素を判別する方法が必要になるが、そのためには、通常は、 α プレーンと呼ばれる透明度を表す同じ画面領域分の情報を別途用意するようにしている。

【0013】しかし、この方法の場合には、描画装置が当該 α プレーンを用いて描画処理をする必要があるため、この α プレーンの情報をテクスチャ画像データと共にメモリに蓄積しておく必要があり、その分のメモリ容量を必要とする。

【0014】このメモリ容量の増大を避けるために、画素値として、透明を表す特別な値、例えば、各画素値を3原色赤（R）、緑（G）、青（B）で表す場合に、 $(R, G, B) = (0, 0, 0)$ のような値、を予め決めておき、描画装置で、画素値がその値のときには、表示画面に応じたフレームバッファの値を書き換えない（つまりテクスチャ画像としては、その画素は透明となる）という処理を行うようにすることも提案されている。

【0015】しかしながら、テクスチャ画像データなどの画像データの高効率圧縮方法は、一般に非可逆的であり、透明部分が誤って不透明として復号される場合があり、その場合には、その不透明画素部分はテクスチャパターンの周辺エッジのノイズとなって表示されてしまう問題がある。

【0016】また、画素値として透明を表す特別の値を用いる方式の場合、不透明画素から急激に描画しない透明画素になるため、テクスチャパターンを縮小・拡大して表示する場合に、テクスチャパターンのエッジに、ぎざぎざしたノイズ、いわゆるエイリアシングノイズが現れることがある。これは、テレビジョン放送技術における、いわゆるクロマキーのブルーバックを用いた画面合成が不完全の場合に生じる現象に近似している。

【0017】この発明は、以上のように、例えばブレンディングされたテクスチャパターンのように、画像として、高画質のものを用意しても、描画時にそれが劣化されて表示されてしまう問題を改善することを目的とする。

【0018】

【課題を解決するための手段】上記課題を解決するために、この発明による画像データ伝送方法は、画素値として透明を表現する値が割り当てられた画像データであって、予め定められた所定の大きさの画面領域分の画像データ単位で、データ圧縮して画像データを伝送すると共に、前記画像データ単位の元の画像データのそれぞれの画素について、それが透明であるか不透明であるかを識別する付加データを、前記データ圧縮された画像データ単位との対応関係をもって伝送することを特徴とする。

【0019】そして、前記データ圧縮された画像データを伸長したときに、前記付加データにより透明画素とされている画素は、伸長画像データに関係なく、強制的に透明画素とすることを特徴とする。

【0020】この画像データ伝送方法によれば、データ圧縮された画像データが伸長復号されたときに、誤差により透明とされるべき画素が不透明となっても、付加データにより、強制的に透明画素に変換される。そして、このように補正された伸長画像データがメモリに蓄えられ、描画装置による描画に用いられる。したがって、付加データをメモリに蓄える必要はなく、付加データの分のメモリ容量は不要であり、しかも、本来の透明部分は透明部分として描画処理されるので、再現画像上においては、非可逆圧縮によるノイズも軽減されることになる。

【0021】また、この発明による画像処理装置は、画素値として透明を表現するための値が割り当てられ、透明を含む描画を行う画像処理装置において、前記透明の画素に隣り合う不透明の画素は、半透明にして描画を行うことを特徴とする。

【0022】この構成の画像処理装置によれば、透明画素の隣りは、半透明の画素に強制的に変換されるため、エイリアシングノイズは軽減される。

【0023】

【発明の実施の形態】以下、この発明による画像処理装置の一実施の形態を、テレビゲーム機の場合について、図を参照しながら説明する。

【0024】図2は、この発明の一実施の形態のテレビゲーム機の構成例を示すもので、この例は3次元グラフィックス機能と、動画再生機能とを備えるゲーム機の場合の例である。

【0025】図3は、この例のゲーム機の外観を示すもので、この例のゲーム機は、ゲーム機本体1と、ユーザの操作入力部を構成するコントロールパッド2とからなる。コントロールパッド2は、このコントロールパッド

2に接続されているケーブル3の先端に取り付けられているコネクタプラグ4を、ゲーム機本体1のコネクタジャック5Aに結合させることにより、ゲーム機本体1に接続される。この例では、いわゆる対戦ゲーム等のために、2個のコントロールパッド2がゲーム機本体1に対して接続することができるように、2個のコネクタジャック5A、5Bがゲーム機本体1に設けられている。

【0026】この例のゲーム機は、ゲームプログラムや画像データが書き込まれた補助記憶手段としての、例えばCD-ROMディスク6をゲーム機本体1に装填することにより、ゲームを楽しむことができる。

【0027】次に、図2を参照しながら、この例のゲーム機の構成について説明する。この例のゲーム機は、メインバス10と、サブバス20とからなる2つのシステムバスを備える構成を有している。これらメインバス10と、サブバス20との間のデータのやり取りは、バスコントローラ30により制御される。

【0028】そして、メインバス10には、メインCPU11と、メインメモリ12と、画像伸長デコード部13と、前処理部14と、描画処理部15と、メインのDMAコントローラ16が接続されている。描画処理部15には、処理用メモリ17が接続されていると共に、この描画処理部15は表示データ用のフレームバッファ（フレームメモリ）と、D/A変換回路を含み、この描画処理部15からのアナログビデオ信号がビデオ出力端子18に出力される。図示しないが、このビデオ出力端子18は、表示装置としての例えばCRTディスプレイに接続される。

【0029】サブバス20には、サブCPU21と、サブメモリ22と、ブートROM23と、サブのDMAコントローラ24と、音声処理用プロセッサ25と、入力部26と、補助記憶装置部27と、拡張用の通信インターフェース部28とが接続される。補助記憶装置部27は、この例ではCD-ROMデコーダ41とCD-ROMドライバ42を備える。ブートROM23には、ゲーム機としての立ち上げを行うためのプログラムが格納されている。また、音声処理用プロセッサ25に対しては、音声処理用メモリ25Mが接続されている。そして、この音声処理用プロセッサ25はD/A変換回路を備え、これよりはアナログ音声信号を音声出力端子29に出力する。

【0030】そして、補助記憶装置部27は、CD-ROMドライバ42に装填されたCD-ROMディスク6に記録されているアプリケーションプログラム（例えばゲームのプログラム）やデータをデコードする。CD-ROMディスク6には、例えば離散コサイン変換（DCT）を用いたMPEG2方式により画像圧縮された動画や静止画の画像データや、ポリゴンを修飾するためのテクスチャ画像の画像データも記録されている。また、CD-ROMディスク6のアプリケーションプログラムに

は、ポリゴン描画命令が含まれている。

【0031】入力部26は、前述した操作入力手段としてのコントロールパッド2と、ビデオ信号の入力端子と、音声信号の入力端子を備えるものである。

【0032】メインCPU11は、メインバス10側の各部の管理および制御を行なう。また、このメインCPU11は、物体を多数のポリゴンの集まりとして描画する場合の処理の一部を行う。メインCPU11は、後述もするように、1画面分の描画画像を生成するための描画命令列をメインメモリ12上に作成する。メインCPU11とメインバス10とのデータのやり取りは、データをバケット形式にしてバケット単位に行い、バースト転送を可能にしている。

【0033】メインメモリ12は、動画や静止画の画像データに対しては、圧縮された画像データのメモリ領域と、伸長デコード処理された伸長画像データのメモリ領域とを備えている。また、メインメモリ12は、描画命令列などのグラフィックスデータのメモリ領域（これをバケットバッファという）を備える。このバケットバッファは、メインCPU11による描画命令列の設定と、描画命令列の描画処理部への転送とに使用される。

【0034】画像伸長デコード部13は、CD-ROMディスク6から再生され、メインメモリに転送された圧縮動画データやメインメモリ12上の圧縮されたテクスチャパターンデータの伸長処理を行なうもので、この例では、MPEG2の画像圧縮方式を採用するので、後述もするように、それに対応したデコーダの構成を有している。

【0035】そして、後述するように、この画像伸長デコード部13の出力段には、瞬時の、つまりはばリアルタイムでの可逆圧縮／伸長が可能で、圧縮率が例えば1/4～1/2程度の瞬時圧縮部50が設けられている。また、この例の画像伸長デコード部13は、その出力画像データの出力形式として、画像データの各画素の値を再量子化して出力する第1の出力データ形式（以下、この第1の出力データ形式をダイレクトカラー形式という）と、前記各画素を、予め定められた限定された数の再現色の中の、当該画素の色が近似する色を示すインデックスデータに変換して出力する第2の出力データ形式（以下、この第2の出力データ形式をインデックスカラー形式という）とを、描画処理部15での処理に適合させて選択可能としている。

【0036】描画処理部15は、メインメモリ12から転送されてくる描画命令を実行して、その結果をフレームメモリに書き込む。フレームメモリから読み出された画像データは、D/A変換器を介してビデオ出力端子18に出力され、画像モニター装置の画面に表示される。

【0037】なお、描画処理部15は、メインメモリ12から受け取る画像データの出力形式がインデックスカラー形式の場合にあっては、各画素データをインデック

ステータから対応する代表色データに変換する処理を行う。このために、描画処理部15は、インデックスデータと代表色データとの変換テーブルであるCLUT (Color Look Up Table) を格納できるようにしている。

【0038】前処理部14は、CPUを備えるプロセッサの構成とされるもので、メインCPU11の処理の一部を分担することができるようにするものである。例えばポリゴンデータを、表示のための2次元座標データに変換する処理も、この前処理部14が行う場合がある。

【0039】そして、この例では、前処理部14とメインバス10との間に、瞬時圧縮部50による瞬時圧縮を解凍する瞬時解凍部60が設けられる。

【0040】次に、このゲーム機の基本的な処理について以下に、簡単に説明する。

【0041】[補助記憶装置部27からのデータの取り込み] 図2の例のゲーム機に電源が投入され、ゲーム機本体1にCD-ROMディスク6が装填されると、ブートROM23の、ゲームを実行するためのいわゆる初期化処理をするためのプログラムが、サブCPU21により実行される。すると、CD-ROMディスク6の記録データが次のようにして取り込まれる。

【0042】すなわち、補助記憶装置部27においては、CD-ROMディスク6から、圧縮画像データ、描画命令及びメインCPU11が実行するプログラムが、CD-ROMドライバ42、CD-ROMデコーダ41を介して読み出され、サブDMAコントローラ24によってサブメモリ22に一旦ロードされる。

【0043】そして、このサブメモリ22に取り込まれたデータは、サブDMAコントローラおよびバスコントローラ30、さらにはメインDMAコントローラ16によってメインメモリ12に転送される。なお、サブCPU21は、描画処理部15のフレームに対して直接的にアクセスできるように構成されており、このサブCPU21によっても表示画像内容の変更が、描画処理部15の制御とは離れて可能とされている。

【0044】[圧縮画像データの伸長及び転送] 図4は、図2のブロック図において、画像データの流れを主体として、より詳細に説明するためのブロック図である。図4において、点線の矢線が画像データの流れを表している。

【0045】図4に示すように、画像伸長デコード部13は、この例では、DMAコントローラ131と、FIFOメモリ132と、MPEGのデコーダ(以下、MDECという)133と、バッカ134と、FIFOメモリ135と、瞬時圧縮部50を備える。瞬時圧縮部50は、瞬時圧縮を行うための変換テーブル52と、DMAコントローラ51とからなる。

【0046】DMAコントローラ131および51は、メインバス10のアービトレーションを行い、メインバ

ス10が空いている時間を縫って圧縮画像データおよび伸長画像データ(瞬時圧縮されている)を、メインメモリ12と、この画像伸長デコード13との間でDMA転送するためのものである。FIFOメモリ132およびFIFOメモリ135は、複数のバス要求が衝突してもデータが失われないようにするための、最低限の段数のバッファである。

【0047】MDEC133は、MPEG2方式の圧縮が施された画像データを伸長デコードする。

【0048】このMDEC133における伸長デコードについて説明する前に、画像データがどのように圧縮されてCD-ROM6に記録されているかについて、図1のブロック図を参照しながら、テクスチャパターンの画像データの場合を例にとって説明する。

【0049】この例では、テクスチャパターンのデータは、前述もしたように、64画素×64画素の矩形領域の2次元画像データであり、各画素データが、赤(R)、緑(G)、青(B)の成分のそれぞれが8ビットからなる。このとき、(R, G, B) = (0, 0, 0)に割り当てられた画素は、透明色と解釈されるように割り当てられている。図5に、プリレンダリングされて得られた高画質の元のテクスチャパターンの例を示す。

【0050】図1に示すように、入力端101を通じて入力されたテクスチャパターンの画像データは、ブロック分割部102において、図5に示すような16画素×16画素の16個の矩形領域に分割される。この矩形領域はマクロブロックと呼ばれる。このブロック分割部102以降、テクスチャパターンのデータは、このマクロブロック単位で処理される。

【0051】この例の場合、ブロック分割部102では、16個のマクロブロックの中で、画素値がすべて透明色なものは、あらかじめ除去されてバックされ、出力される。同時に、テクスチャパターン内におけるマクロブロックの位置情報を示すテーブルMtbが、ブロック分割部102で生成され、記録信号生成部108に供給されて、テクスチャパターンデータの付加情報であるヘッダ情報に含めて記録されるようにされる。

【0052】このマクロブロックの位置テーブルPtbは、例えば図5のようなテクスチャパターンの場合には、図6に示すようなものとなる。すなわち、マクロブロックの位置テーブルPtbは、すべて透明色の画素からなるマクロブロック位置に対しては「0」、マクロブロック内に不透明の画素が1画素でも存在する場合には「1」が割り当てられた4×4×1ビットのテーブルである。この例の場合には、画素値がすべて透明色のマクロブロックが除去されてバックされたテクスチャパターンの出力画像データは、図7に示すように10個のマクロブロックからなるものとなる。

【0053】このバックされたテクスチャパターンのデ

ータは、圧縮部103に供給される。この圧縮部103では、各マクロブロックは、図8に示すように、3原色信号表現から、輝度信号と色差信号とからなる表現に変換される。これを、以下、CSC (Color Space Conversion) と呼ぶことにする。この際に用いられるCSC係数の例を図9の表1に示す。

【0054】マクロブロックは、輝度信号成分については、図8に示すように、4分割されて、それぞれ8画素×8画素の領域からなる4個の輝度ブロックY0、Y1、Y2、Y3により構成される。また、色差信号成分からなるマクロブロックは、それぞれ隣り合う4画素がまとめられて、8画素×8画素のブロックにまとめられた2つの色差信号ブロックから構成される。こうして、マクロブロックは、合計6個のブロックに分けられる。

【0055】このマクロブロックに対して、圧縮部103でDCT (Discrete Cosine Transformation) がかけられる。DCTは、一般的には直交変換と呼ばれる相似変換の一種で、ブロックの各輝度値を成分とする8画素×8画素の行列Xを考えると、DCT行列P、その逆行列Piを用いて、 $Y = P \cdot X \cdot P_i$

の形で定義される変換を指す。上記DCT行列Pの係数は、図10に示す表2の通りである。

【0056】DCT変換されたブロックは成分ごとに異なる分解能で量子化される。成分ごとの量子化の幅(ステップ)を指定するテーブルを量子化テーブル(Qテーブル)と呼ぶ。その量子化テーブルの例を図11の表3に示す。

【0057】実際の量子化は成分ごとに対応するQテーブルの値に、全体の量子化ステップを決める値QUANTの積で割ることによって行なわれる。

【0058】ここで、全体の量子化ステップQUANTの値を大きくするとデコード後の画質は劣化するが、ブロック内の0成分の個数が多くなるため、圧縮率を向上することができる。

【0059】量子化されたブロックはジグザグオーダーと呼ばれる順序で1次元に番号づけされた後、ハフマン符号化による可変長符号化が行われる。そして、この圧縮部103よりの圧縮画像データは、記録信号生成部109に供給される。

【0060】ブロック分割部102からのバックされたテクスチャパターンの画像データは、また、マスクパターン生成部104に供給され、バックされたマクロブロックごとに、透明色に対応するビットが1になる、16×16×1ビットのαパターン(マスクパターン)が用意され、これがマスクパターンバッファ105を介して記録信号生成部109に供給される。

【0061】例えば、テクスチャパターンの一つのマクロブロックが、図12に示すようなパターンを有する場合、そのマクロブロックのマスクパターンMaskは、図

13に示すようなものとなる。なお、図12において、右側に示したのは、各画素値を8ビット表現したときのレベル値であり、画素値=0は、前述したように、3原色からなる画素値が(R, G, B) = (0, 0, 0)に相当し、透明を意味するものである。マスクパターンMaskのデータは、透明である画素に対して「0」、不透明である画素に対して「1」が設定されたものであり、16×16×1ビットのデータである。

【0062】また、この例では、インデックスカラー方式のデータ処理を行うためのCLUT(色変換テーブル)が生成され、このCLUTのデータも、CD-ROM6に記録される。このため、ブロック分割部102からのマクロブロック単位の画像データは、CLUT生成部106に供給される。このCLUT生成部106では、ベクトル量子化により、マクロブロック単位で、そのマクロブロックで使用すべき代表色を選定する。そして、その代表色とインデックスとからなる図14に示すようなCLUTを生成する。

【0063】CLUT生成部106でのベクトル量子化の手法としては、例えば、赤、緑、青の原色信号成分を互いに直交する方向にとって3次元空間を考え、各画素間のその色空間上の距離を求め、互いに距離の短い画素同志をまとめることにより、マクロブロック内の画素の色が、この例では、16色の代表色以内に収まるように画素データを丸める方法を採用する。なお、この方法のほかにも、従来から提案されている種々のベクトル量子化の手法を用いることももちろんできる。しかし、この実施の形態においては、いずれの方法を用いる場合であっても、透明色の画素がマクロブロック内に含まれている場合には、代表色の一つは必ず、透明色が割り当てられるように構成される。

【0064】こうして16色、あるいはそれ以下に代表色を丸めることができたなら、その16色あるいはそれ以下の代表色と、各代表色のインデックスとからなる、前述した図14に示す色変換テーブル、つまりCLUTを作成する。そして、作成したCLUTは、CLUTバッファ107を介して記録信号生成部108に供給する。

【0065】記録信号生成部108は、以上の結果をマクロブロック単位にまとめ、書き込み部109を通じてCD-ROM6に書き込むようにする。この場合、例えば、図15に示すように、マクロブロック単位の圧縮画像データの次にそのマクロブロックのマスクパターンMaskおよびCLUTのデータを挿入するようなデータの形式となり、これをビットストリームと呼ぶ。すなわち、マクロブロック単位の圧縮画像データとマスクパターン等との組みの連続が、64画素×64画素分のテクスチャパターンのデータとされ、これがCD-ROM6に記録されている。なお、一つのテクスチャパターンのデータの先頭には、ヘッダ情報が付加される。このヘッ

タ情報には、テクスチャパターンのデータであることを示す識別データのほか、前述したように、テクスチャパターン内でのマクロブロックの位置情報を示す図6に示したマクロブロック位置テーブルPtbが含まれて記録される。

【0066】以上のようにして圧縮されてCD-ROM 6に記録されている画像データの画像伸長デコードの手順は、上記の画像圧縮の手順の逆を辿ることで行なわれる。この場合の圧縮は非可逆であるので、画像伸長デコード部13では、後述もするように、伸長したマクロブロックの透明色画素が正しく復号されるように、画素値の圧縮データにインターリーブされて記録されている、マスクパターンMaskを使用して、復号後の画素で対応するマスクパターンMaskのビットが「0」の画素は、復号結果の画素値に関わらず強制的に透明色に変更される。

【0067】画像伸長デコード部13のMDEC133は、以下の処理により構成される。

(1) CD-ROM 6には、圧縮画像および付加情報（ジオメトリ情報など）がインターリーブして記録されている。これらの情報は、CD-ROMドライバ42およびCD-ROMデコーダ41により、連続的に読み出され、前述したように、一旦、メインメモリ12に格納された後に、圧縮された画像情報のみが切り出されて、画像伸長デコード部13に転送されてくる。この圧縮画像情報には、マスクパターンMaskが付随している。なお、ジオメトリ情報などの付加情報は、CPU11において処理されて、これにより解凍画像がテクスチャとして使用されるオブジェクトの位置情報が計算される。

(2) MDEC133は、可変長復号化器を備え、ハフマン符号化されたブロックを復号する。このときのハフマンの木は固定であるが、対応する符号の値は変更できる。

(3) MDEC133は、また、逆量子化器を備える。この逆量子化器は、復号されたブロックに逆量子化をかけ、前述したジグザグオーダーに対応してブロック順序を変換する。逆量子化はブロックの各係数単位に異なるステップで行なわれる。

(4) MDEC133が備える逆DCT変換器は、8×8画素の逆直交変換を行なう。

(5) MDEC133は、さらにCSC処理部を備え、輝度信号・色差信号で表現されたマクロブロック画像を3原色信号R、G、Bで表現されたものに変換する。

【0068】以上のようにして、補助記憶装置部27からメインメモリ12に転送された入力データのうち、圧縮画像データは、DMAコントローラ131によりメインメモリ12から画像伸長デコード部13に転送され、MDEC133で、MPEG2に対応したデコード処理が行われて、圧縮された画像データが解凍され、各画素が上述のような3原色信号R、G、Bからなるダイレク

トカラー形式の画像データとしてデコードされる。

【0069】この画像データは、バッカー134に供給される。このバッカー134は、伸長デコードされた画像データを、その画素単位で描画処理部15に適合する形式に梱包するものである。すなわち、前述したように、この例においては、描画処理部15に送る画像データの出力形式をダイレクトカラー形式と、インデックスカラー形式とのいずれかにすることができ、この出力形式の変換をバッカー134は行う。また、このバッカー134は、マクロブロック単位のデータについて、マスクパターンMaskを用いて、非可逆圧縮／伸長により、透明画素が不透明画素に変化していても、それを強制的に透明画素に変換する処理も行う。

【0070】このバッカー134は、図16に示すような構成を有する。すなわち、バッカー134は、ディザマトリクステーブル72に基づいてディザ処理を行うディザ処理部71と、インデックスカラー形式に変換するために、画素データをCLUT格納部74に用意されるCLUTの代表色に纏めるようにするベクトル量子化器73と、バック処理部75と、マスクパターンMaskを格納するマスクパターン格納部76とからなる。

【0071】CLUT格納部74には、テクスチャパターン画像データにインターリーブされて付加されている各マクロブロック単位のCLUTが、メインメモリ12から転送されて格納される。また、マスクパターン格納部76には、同様にして、テクスチャパターン画像データにインターリーブされて付加されている各マクロブロック単位のマスクパターンMaskが、メインメモリ12から転送されて格納されている。なお、これらCLUTやマスクパターンMaskは、メインメモリを経由せずに、CD-ROM 6から直接的に、格納部74および76に転送するようにすることもできる。

【0072】バック処理部75では、復号データを画素単位にバッキングして出力する。この際に、マスクパターン格納部76に格納されているマスクパターンMaskを用いて、圧縮前のもとのテクスチャ画像で、透明であった画素の伸長データを強制的に透明色の値に変更する処理を行う。

【0073】例えば、圧縮前の元のマクロブロックのパターンが、図12のようなもので、そのマスクパターンMaskが図13に示すようなものであった場合に、MPEG2による非可逆圧縮／伸長により、伸長デコードされたマクロブロックのパターンが図17に示すように、透明であった画素が不透明の画素として伸長されたものとなった場合を考える。

【0074】この場合、バック処理部75には図17に示すようなマクロブロックのデータが入力されるが、マスクパターン格納部76には、図13に示したマスクパターンMaskが格納されているので、バック処理部75では、図13のマスクパターンMaskで「0」となって

いる画素は、入力伸長画像データの画素値の如何に関係なく、すべて透明色であることを示す(R, G, B) = (0, 0, 0)に変更する。これにより、バック処理部75からは図18に示すように、もともと透明色であった画素は、すべて正しく透明色とされた伸長画像データが得られる。

【0075】バッカー134から、伸長画像データを、ダイレクトカラー形式で出力する場合であって、入力画素のビット数と、出力画素のビット数が等しいときには、ディザ処理部71およびベクトル量子化器73はバイパスされ、復号データがバック処理部75において画素単位にパッキングされて出力される。

【0076】ダイレクトカラー形式で出力する場合であっても、入力画素のビット数よりも、出力画素のビット数Nが小さいときには、ディザ処理部71において適当な丸め処理が行われる。この例の場合、MDEC133からは16ビットの符号付き固定小数点形式で復号データが得られるもので、丸め処理は、以下のうちのいずれかとされる。

a) 入力画素がNビットの範囲に収まるようにクリップ処理がされた後、入力画素値の整数部の下位Nビットが出力される。

b) 入力画素の上位N+1ビットが四捨五入され、その上位Nビットが出力される。

c) 入力画素に固定のディザマトリクステーブル72のオーダーディザが掛けられた後に、上位N+1ビットが四捨五入され、その上位Nビットが出力される。

【0077】次に、インデックスカラー形式で出力する場合には、前述したようにしてCLUT格納部74に格納された図14に示したようなCLUTの代表色を用いた逆ベクトル量子化が、ベクトル量子化器73において*

$$D = (R1 - R2) * (R1 - R2) + (G1 - G2) * (G1 - G2) + (B1 - B2) * (B1 - B2) \dots (Q1)$$

この式(Q1)で、*は、掛け算を示している。

【0084】次のステップ206で、求めた距離Dと最小距離Dminの値の比較を行い、求めた距離Dがそれまでの最小距離Dminよりも小さければ、ステップ207において、最小距離Dminを求めた距離Dに置き換えると共に、求めるインデックス値kminを、その代表色の欄のポインタ値kに置き換える。一方、求めた距離Dがそれまでの最小距離Dminよりも大きければ、ステップ208に進み、最小距離Dminおよび求めるインデックス値kminをそれまでの値のままとする。

【0085】ステップ207およびステップ208の次にはステップ209に進み、CLUTのすべての代表色を参照したか否か判断し、未だ参照がされていない代表色が残っていれば、ステップ210に進み、ポインタkの値をインクリメントし、ステップ204に戻って、上述のステップ204からステップ209までの動作を繰り返す。

【0086】ステップ209で、当該画素について、C

*行われ、各画素値の代わりに、対応する代表色のインデックスのデータが出力される。

【0078】ベクトル量子化の手法としては、例えば、伸長デコード後の画素値(3原色R, G, Bのそれぞれの色成分について、例えば8ビットからなる色データ)と、CLUTとして与えられた16種類の代表色データ(3原色R, G, Bのそれぞれの色成分について、例えば4ビットからなる)とを比較して、CLUT74中で、最も、色が近い色データのインデックスを画素値の代わりに出力する。

【0079】図19は、このベクトル量子化器3の処理動作のフローチャートである。

【0080】まず、ステップ201において、伸長デコードされたマクロブロックのデータの最初の画素を取り出し、次のステップ202でその画素Px, yの画素値(R, G, B)を読む。次に、ベクトル量子化のための初期化を行う。この初期化は、CLUTの参照欄をポインタk(ここで、kは図14におけるインデックス番号に等しい)を初期値にすると共に、CLUTの代表色データと画素Px, yの画素値との距離D(R, G, B色空間上の距離、つまり、色の近似の度合い)の最小値Dminを所定の大きい値にする。

【0081】次に、ステップ204に進み、ポインタkで示されるCLUTの代表色データを参照し、ステップ205で、その参照代表色データと、画素Px, yとの距離Dを求める。

【0082】この場合、2つの色(R1, G1, B1)と(R2, G2, B2)との距離Dは、例えば以下の式で計算される。

【0083】

CLUTのすべての代表色の参照が終了したと判断されたときには、ステップ211に進み、インデックス値kminを、その画素のデータとして出力する。そして、ステップ212に進み、マクロブロック内のすべての画素について、上述のCLUTを用いたインデックスデータへの変換が終了したか否か判断し、終了していない次の画素があれば、ステップ213に進んで、当該次の画素を読み出した後、ステップ202に戻り、その画素についての上述の処理を行う。また、マクロブロック内のすべての画素についての処理が終了したと判断したときには、次のマクロブロックの処理に移る。

【0087】このようにして、圧縮伸長デコード部13は、インデックスカラー方式では、伸長デコード後の24ビットの画素値を、この例では、4ビットのインデックスのデータに圧縮変換して出力する。この伸長出力画像データには、マスクパターンMaskのデータやCLUTのデータは、含まれていないことはいうまでもない。

【0088】以上のようにして、バッカー134で梱包

された画素データは、FIFOメモリ135を通じて瞬時圧縮部50に送られ、画像データが瞬時圧縮される。ここでの瞬時圧縮は、MPEG2のような高能率圧縮とは異なり、 $1/4 \sim 1/2$ と圧縮率は低い、ハードウェア規模が小さな圧縮/復号回路で高速に可逆的に圧縮/復号ができるものが使用される。

【0089】この例の場合には、圧縮にはランレングス符号化とハフマン符号化が同時に行なわれる。この圧縮のための辞書としてのコードブックである変換テーブル52が、瞬時圧縮部50に設けられる。このコードブックである変換テーブル52の生成は、予め行われて、保持されている。

【0090】図示のように、瞬時圧縮部50は、機能的にDMAコントローラ51を備え、変換テーブル52を用いて、ランレングス符号化とハフマン符号化が同時に行なって瞬時圧縮しながら、MPEG伸長デコードされた画像データを、メインメモリ12に転送する。以上が、画像伸長デコード部13の動作である。この場合、圧縮後のデータは、透明色として特定の値が与えられた画像データであるので、メインメモリ12には、透明度を表す α プレーンを記憶することは不要であり、その分だけ、メモリ容量が少なくて済む。しかも、この実施の形態では、瞬時圧縮部50により、圧縮されたデータがメインメモリ12に記憶されるため、データ量が少なくなる。

【0091】なお、動き補償を行なう場合は、瞬時圧縮部50での瞬時可逆圧縮を行なわない。この場合、画像伸長デコード部13はビットストリームを読み込む際に、同時にメインメモリ12上に展開された前フレームの画像データを同時に読み込んで処理を行なう。

【0092】〔描画命令列についての処理と転送〕物体の面を構成するポリゴンは、3次元的な奥行き情報であるZデータに従って奥行き方向の深い位置にあるポリゴンから順に描画することにより、2次元画像表示面に立体的に画像を表示することができる。メインCPU11は、このように奥行き方向の深い位置にあるポリゴンから順に、描画処理部15で描画が行われるようにするための描画命令列をメインメモリ12上に作成する。

【0093】メインCPU11は、入力部26のコントロールパッドからのユーザーの操作入力に基づいて、物体や視点の動きを計算し、メインメモリ12上にポリゴン描画命令列を作成する。

【0094】この描画命令列が完成すると、メインDMAC16は、前処理部14を通じて、描画命令毎に、メインメモリ12から描画処理部15に転送する。

【0095】描画処理部15では、送られてきたデータを順次実行して、その結果を、フレームメモリの描画領域に格納する。このポリゴン描画の際、データは、描画処理部15の勾配計算ユニットに送られ、勾配計算が行なわれる。勾配計算は、ポリゴン描画で多角形の内側を

マッピングデータで埋めていく際、マッピングデータの平面の傾きを求める計算である。テクスチャの場合はテクスチャ画像データでポリゴンが埋められ、また、グローシェーディングの場合は輝度値でポリゴンが埋められる。

【0096】更に、動画のテクスチャが可能である。つまり、動画テクスチャの場合には、前述したように、CD-ROMディスク6からの圧縮された動画データは、一旦、メインメモリ12に読み込まれる。そして、この圧縮画像データは、画像伸長デコード部13に送られる。画像伸長デコード部13で、画像データが伸長される。

【0097】そして、前述したようにして、伸長された動画データは描画処理部15のフレームメモリ上のテクスチャ領域に送られる。テクスチャ領域は、この描画処理部15のフレームバッファ内に設けられているので、テクスチャパターン自身も、フレーム毎に書き換えることが可能である。このように、テクスチャ領域に動画を送ると、テクスチャが1フレーム毎に動的に書き換えられて変化する。このテクスチャ領域の動画により、ポリゴンへのテクスチャマッピングを行えば、動画のテクスチャが実現される。

【0098】メインCPU11は、伸長されたマクロブロック単位の画像データが一定量、メインメモリ12に蓄積された時点で、当該伸長データを、メインバス10を通じ、瞬時解凍部60および前処理部14を介して描画処理部15のフレームバッファに転送するようにする。この際に、伸長画像データがフレームバッファの画像メモリ領域に転送されれば、そのまま背景動画として画像モニター装置で表示されることになる。また、フレームバッファのテクスチャ領域に転送されれば、このテクスチャ領域の画像データは、テクスチャ画像として、ポリゴンの修飾に使用される。

【0099】瞬時解凍部60は、機能ブロックとしてのDMAコントローラ61と、瞬時圧縮部50の変換テーブル52とは逆変換を行うための変換テーブル62とからなり、メインメモリ12からの瞬時圧縮されている画像データを、変換テーブル62を用いて解凍し、MPEG伸長デコードされた画像データとし、前処理部14を通じて描画処理部15に送る。

【0100】この例の場合、出力形式がダイレクトカラー形式の場合には、前処理部14から、各画素のデータが指定された所定のビット数のR、G、Bの3原色信号からなる画像データが描画処理部15に供給されて、描画処理が実行される。

【0101】一方、インデックスカラー形式の場合には、描画処理部15には、前述したようなインデックスデータが供給される。描画処理部15は、前述のCLUT格納部74に格納されるものと同様のCLUTがメインメモリ12から転送されて格納されており、描画処理

部15は、このCLUTを用いて、インデックスカラー形式の画像データを、対応する代表色データに変換する処理を行い、画像データを復元する。そして、この復元した画像データを用いて描画処理を実行する。

【0102】そして、描画処理部15では、上述のようなテクスチャパターンを用いたテクスチャマッピング処理を行う場合において、透明の画素に隣り合う不透明の画素は、半透明にして描画する処理をも行う。

【0103】図20は、描画処理部15で実行される半透明処理の一例のフローチャートである。この例の半透明処理においては、画像の水平方向について、透明画素の隣りは半透明とするものである。

【0104】すなわち、まず、ステップ301において、マクロブロックのデータの最初の画素を取り出し、次のステップ302でその画素 P_x, y の画素値を読む。この画素値は、インデックスデータである場合には、CLUTから得た代表色データである。次に、ステップ303において、当該画素が透明であるか否か判断する。透明であるか否かは、前述したように、画素値が、予め設定された透明色 $(R, G, B) = (0, 0, 0)$ であるか否かにより判断する。

【0105】当該画素 P_x, y が透明であれば、描画処理部15では、ステップ307に進み、当該画素 P_x, y は透明画素として描画する。すなわち、フレームバッファの当該画素 P_x, y の位置の画素データは書き換えず、背景の色のままとする。

【0106】ステップ303で、画素 P_x, y が不透明であると判別されたときには、ステップ304に進み、当該画素 P_x, y がマクロブロックの水平方向の端部の画素であるか否か判断し、水平方向の端部の画素であれば、ステップ309に進み、その画素 P_x, y は、半透明の画素として描画する。すなわち、フレームバッファの当該画素 P_x, y の位置の画素データは、背景の色と当該画素 P_x, y の色の1:1の混合とする。

【0107】ステップ304で、当該画素 P_x, y がマクロブロックの水平方向の端部の画素でないと判別されたときには、ステップ305に進み、当該画素 P_x, y に対し、水平方向に一つ手前の隣接画素 P_{x-1}, y が透明であるか否か判断する。隣接画素 P_{x-1}, y が透明であれば、ステップ309に進み、当該画素 P_x, y は、半透明の画素として描画する。

【0108】ステップ305で、隣接画素 P_{x-1}, y が透明でないと判別されたときには、ステップ306に進む。ステップ306では、当該画素 P_x, y の水平方向に一つ後の隣接画素 P_{x+1}, y が透明であるか否か判断する。隣接画素 P_{x+1}, y が透明であれば、ステップ309に進み、当該画素 P_x, y は、半透明の画素として描画する。

【0109】ステップ306で、隣接画素 P_{x+1}, y が透明でないと判断されたときには、画素 P_x, y に対

して水平方向に隣接する2個の画素は共に不透明であるので、ステップ308に進んで、当該画素 P_x, y は、不透明の画素として描画する。すなわち、フレームバッファの対応する画素値を当該画素 P_x, y の値に書き換える。

【0110】今、当該画素 P_x, y の混合率を α とすると、 $\alpha = 0$ 、0は透明、 $\alpha = 0.5$ は半透明、 $\alpha = 1$ 、0は不透明の描画処理となる。上述の半透明処理を、図12に示したようなマクロブロックのパターンの例について適用した場合の結果の例を示すと、図21のようなものとなる。これにより、テクスチャマッピングにおけるエイリアシングノイズを軽減することができる。

【0111】なお、以上の半透明処理においては、水平方向の隣接画素のみが透明であるときに、不透明画素を半透明に描画するようにしたが、垂直方向の隣接画素のみを参照して半透明にするか否かを決定するようにしてもよい。また、水平方向および垂直方向のいずれかの隣接画素が透明であるときに、当該不透明画素を半透明で描画するようにしてもよい。さらには、斜め方向の隣接画素をも参照して半透明処理をするようにしてもよい。

【0112】また、単に隣接する画素が透明であるかにより半透明の描画を行うのではなく、例えば、水平方向または垂直方向に連続して透明画素が続いている状態から、不透明画素に変化したときの最初の不透明画素を半透明で描画し、また、不透明画素が連続している状態から、透明画素に変化したとき、透明画素への変化直前の不透明画素を半透明で描画するようにしてもよい。

【0113】以上のようにして、この実施の形態によれば、透明色の画素に対しては予め特定の画素値を与えた画像データに対して、非可逆圧縮方式により高能率に圧縮するが、透明色の画素と不透明の画素を認識するために、1画素あたり1ビットからなるマスクパターン Msk を付加データとして、圧縮画像データに付加するようにし、圧縮画像データを伸長したときに、このマスクパターン Msk を用いて、本来透明である画素は強制的に透明画素データに変更するようにするので、透明画素は常に正しく透明の描画がなされる。

【0114】また、描画のためには、 α プレーンは不要であるので、メインメモリ12の容量がその分、少なくともすむ。

【0115】また、透明画素に隣接する不透明画素は、半透明に描画するようにすることにより、描画画像の輪郭部分のエイリアシングノイズを軽減することができる。

【0116】また、この実施の形態によれば、伸長画像データは、瞬時圧縮されてメインバス10を通じてメインメモリに転送される。したがって、圧縮されている分だけ、メモリの利用効率が向上する。しかも、伸長画像データの出力形式としてインデックスカラー形式とした場合には、画素データがインデックスのデータで構成さ

れるために、データ量が少なくなり、その分だけ、メインメモリの利用効率がよくなる。

【0117】また、メインバス10を通じて画像伸長デコード部13からメインメモリ12に転送される伸長画像データおよび、メインメモリ12からメインバス10を通じて描画処理部15に転送される伸長画像データは、それぞれ瞬時圧縮されているデータであり、データ量が少ないので、バスの転送速度が向上する。

【0118】また、この実施の形態では、入力圧縮画像データの形式が1種類であっても、画像伸長デコード部13からの出力データの出力形式をダイレクトカラー形式とインデックスカラー形式とのいずれかを選択することが可能であり、別々の出力形式を得るために、入力画像データを別途用意する必要がないので、その点でも、メインメモリの利用効率が向上する。

【0119】また、ダイレクトカラー形式の場合においても、ディザ処理により、描画処理部15での処理に適したビット数に丸めることができるので、所望のビット数の出力データを容易に得ることができる。

【0120】なお、以上の例は、この発明による画像処理装置をゲーム機に適用した場合であるが、この発明による画像処理装置は、種々の用途に使用できることはいうまでもない。

【0121】

【発明の効果】以上説明したように、この発明によれば、透明画素については常に正しく透明の描画を行うことができる。しかも、透明の描画処理時には、 α ブレインのような付加的なデータを用いないので、このような付加的なデータをメモリに保存する必要がない。

【0122】また、透明画素に隣接する不透明画素は、半透明の描画処理をすることにより、エイリアシングノイズを軽減することができる。

【図面の簡単な説明】

【図1】この発明による画像データ伝送方法の一実施の形態を説明するための記録処理部の例を示すブロック図である。

【図2】この発明による画像処理装置の一実施の形態としてのゲーム機の構成例を示すブロック図である。

【図3】図2の例のゲーム機の外観例を示す図である。

【図4】この発明の一実施の形態における圧縮画像データの処理の流れを説明するための図である。

【図5】この発明の一実施の形態における元の画像およびその処理単位を説明するための図である。

【図6】図5の画像の処理に用いる付加的なデータを説明するための図である。

【図7】図5の元の画像を記録するときのデータを説明するための図である。

【図8】この発明の一実施の形態における圧縮画像データの処理単位を説明するための図である。

【図9】この発明の一実施の形態における圧縮画像データの圧縮方式を説明するための図である。

【図10】この発明の一実施の形態における圧縮画像データの圧縮方式を説明するための図である。

【図11】この発明の一実施の形態における圧縮画像データの圧縮方式を説明するための図である。

【図12】この発明の一実施の形態における圧縮画像データの処理単位の元の画像パターンの例を示す図である。

【図13】図12の画像パターンに付随する付加的なデータを示す図である。

【図14】インデックスカラー形式のための色変換テーブルを説明するための図である。

【図15】この発明の一実施の形態における圧縮画像データの記録時のフォーマットの例を示す図である。

【図16】図2の一部のブロックの詳細例を示すブロック図である。

【図17】非可逆圧縮より、画像データに生じるノイズを説明するための図である。

【図18】この発明の一実施の形態により、非可逆圧縮により生じたノイズを除去した後の画像パターンを示す図である。

【図19】インデックスカラー形式のデータに変換する処理のフローチャートの例を示す図である。

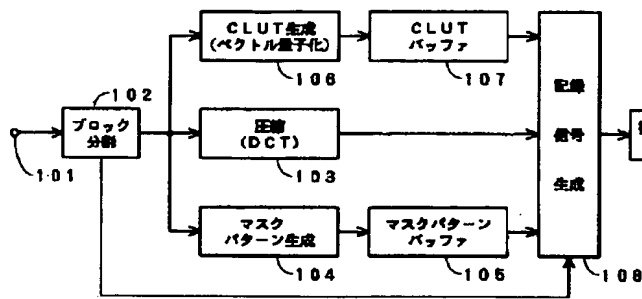
【図20】半透明処理を含む描画処理動作のフローチャートの例を示す図である。

【図21】半透明処理を説明するための図である。

【符号の説明】

10…メインバス、11…メインCPU、12…メインメモリ、13…画像伸長デコード部、14…前処理部、15…描画処理部、16…メインDMAコントローラ、50…瞬時圧縮部、51…瞬時圧縮用の変換テーブル、60…瞬時解凍部、61…圧縮解凍用の変換テーブル、71…ディザ処理部、72…ディザマトリクステーブル、73…ベクトル量子化器、74…CLUT格納部、75…バック処理部、76…マスクパターン格納部、102…ブロック分割部、103…画像データの圧縮部、104…マスクパターン生成部、106…CLUT生成部、108…記録信号生成部、132および135…FIFOメモリ、133…MPEGのデコーダ、134…バッカー

【図1】

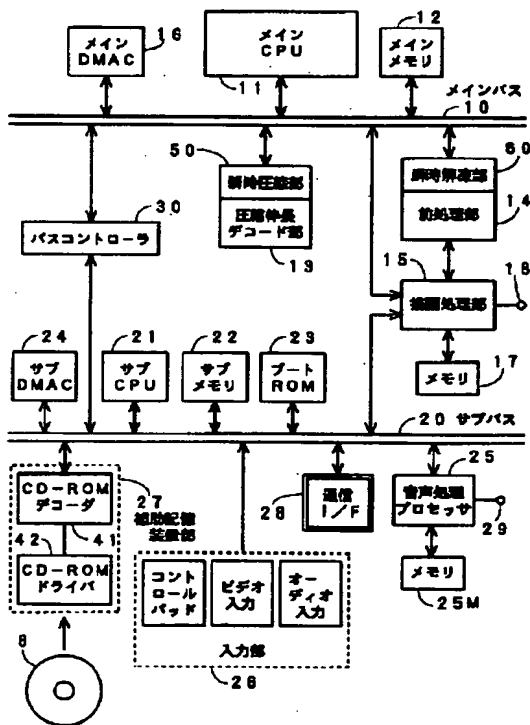


【図6】

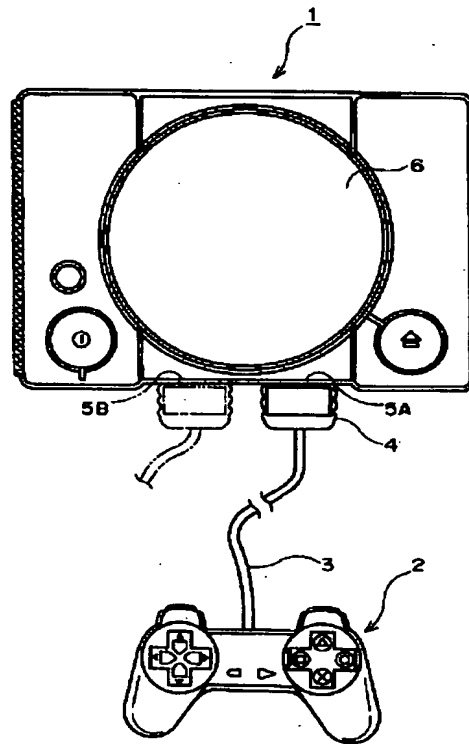
0	1	1	0
0	1	1	1
1	1	1	1
1	1	0	0

Ptb
マクロブロック
位置テーブル

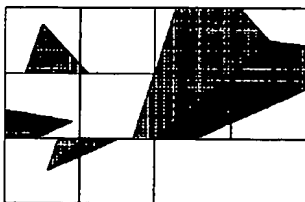
【図2】



【図3】



【図7】



バック側のテクスチャパターン

【図9】

表1 C9C係数の例

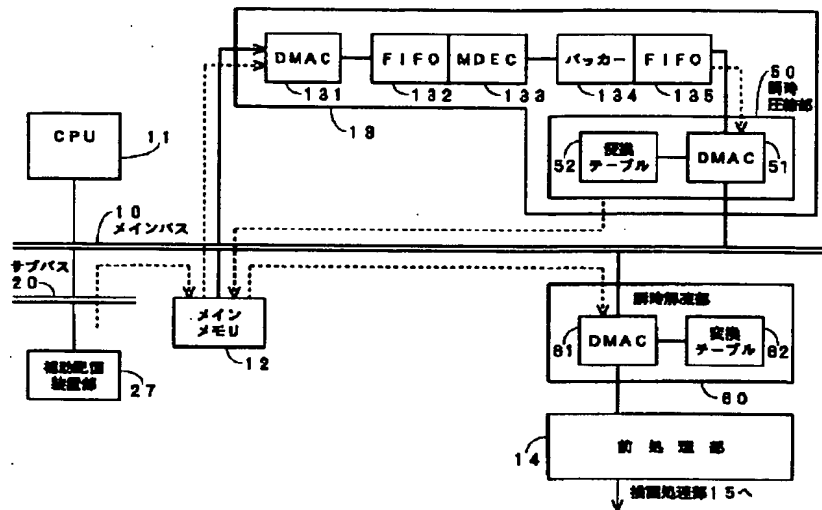
$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299 & 0.567 & 0.114 \\ -0.16871 & -0.33130 & -0.5 \\ 0.5 & -0.4187 & -0.0813 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

【図11】

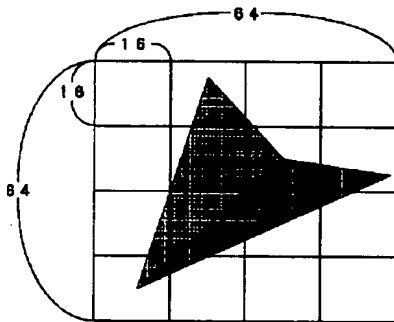
表3 量子化テーブルの例

$$Q_{tab} = \begin{bmatrix} 2 & 16 & 18 & 22 & 26 & 27 & 29 & 34 \\ 16 & 16 & 22 & 24 & 27 & 29 & 34 & 37 \\ 19 & 22 & 25 & 27 & 29 & 34 & 34 & 36 \\ 22 & 22 & 25 & 27 & 29 & 34 & 37 & 40 \\ 22 & 25 & 27 & 29 & 32 & 35 & 40 & 48 \\ 26 & 27 & 29 & 32 & 35 & 40 & 48 & 56 \\ 26 & 27 & 29 & 34 & 36 & 40 & 56 & 63 \\ 27 & 29 & 35 & 36 & 40 & 56 & 63 & 63 \end{bmatrix} \times 1/16$$

【図4】

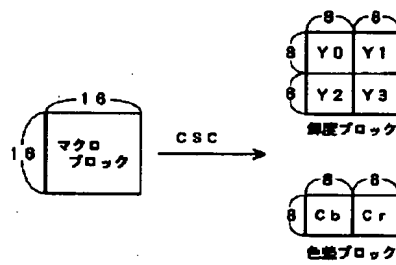


【図5】



テクスチャパターン (ソース)

【図8】



【図13】

【図10】

表2 DCT係数の例

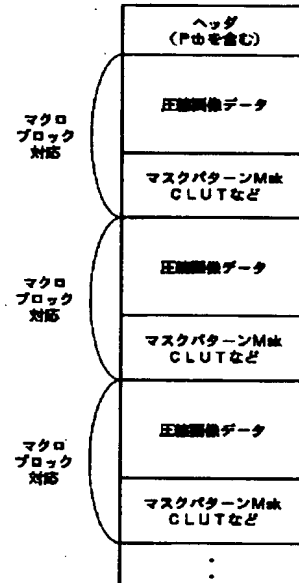
P =	4096	4096	4096	4096	4896	4096	4096	4096
	5681	4816	3218	1130	-1130	-3218	-4816	-5681
	5332	2217	-2217	-5332	-5332	-2217	2217	5332
	4816	-1120	-5681	-3218	3218	5681	1130	-4816
	4096	-4096	-4096	4096	4896	-4096	-4096	4096
	3218	-5681	1120	4816	-4816	-1130	5681	-3218
	2217	-5332	5332	-2217	-2217	5332	-5332	2217
	1130	-3218	4816	-5681	5681	-4816	3218	-1130

×1/64

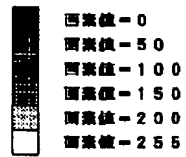
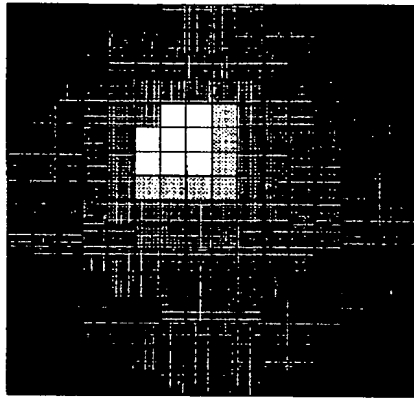
0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	0	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	0	0	0
0	0	1	1	1	1	1	1	1	1	1	1	1	0	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	0	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0
0	0	0	0	0	0	1	1	1	1	0	0	0	0	0

マスクパターンMsk

【図15】

テクスチャパターンの
記憶データ

【図12】

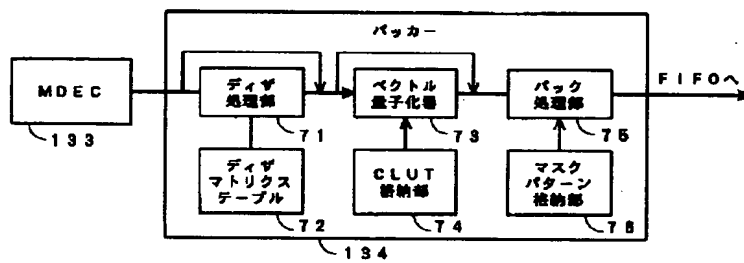


【図14】

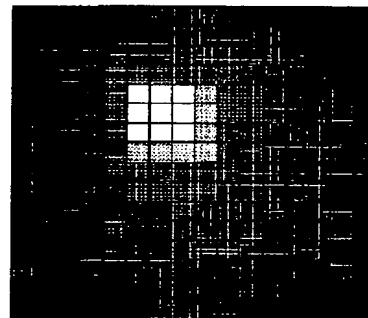
CLUT

インデックス (色番号, 4ビット)	代表色データ (R, G, B 合計で24ビット)
0	色データ 0
1	色データ 1
2	色データ 2
⋮	
15	色データ 15

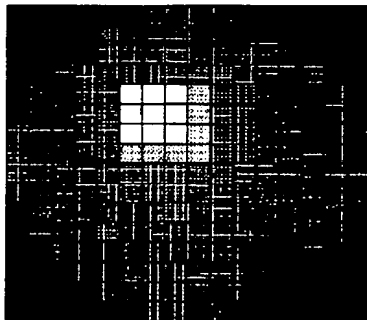
【図16】



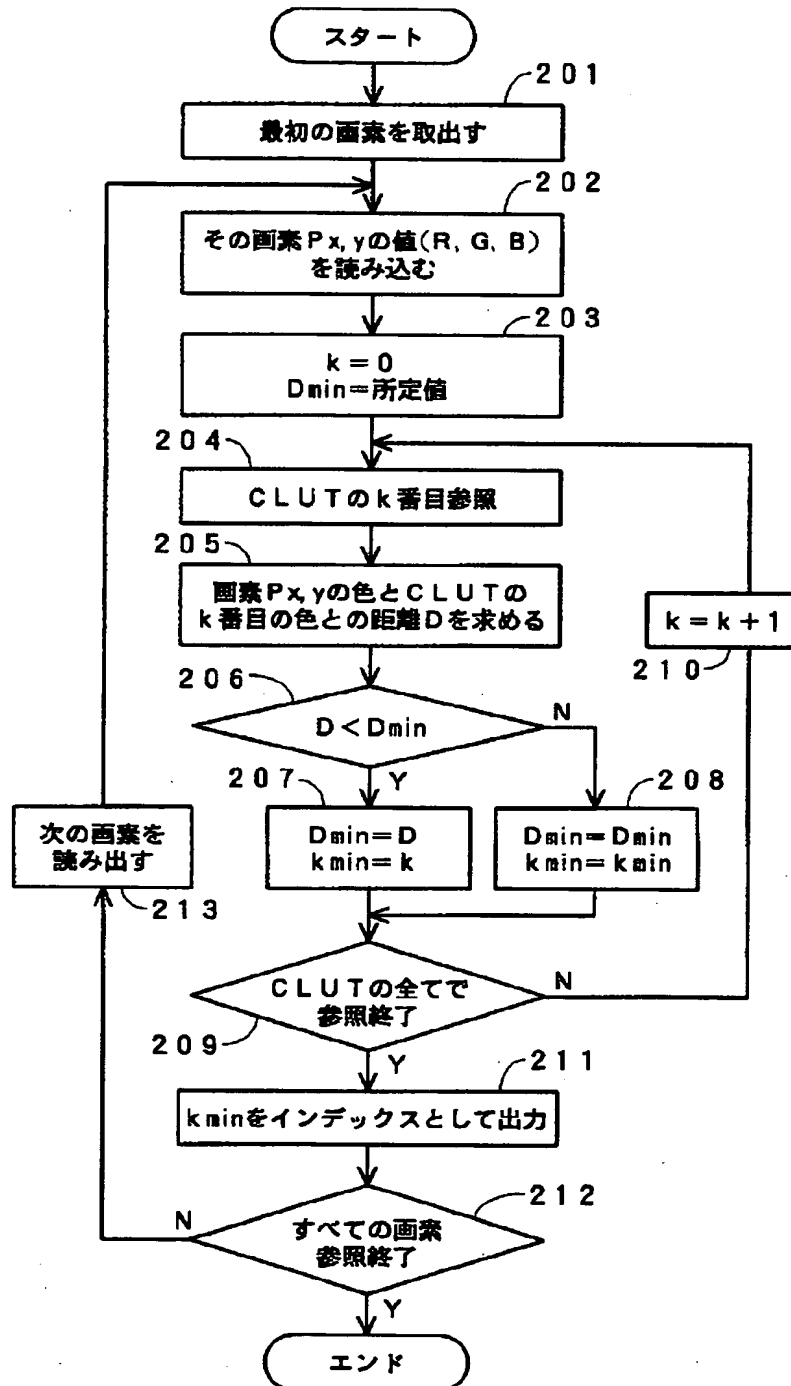
【図17】



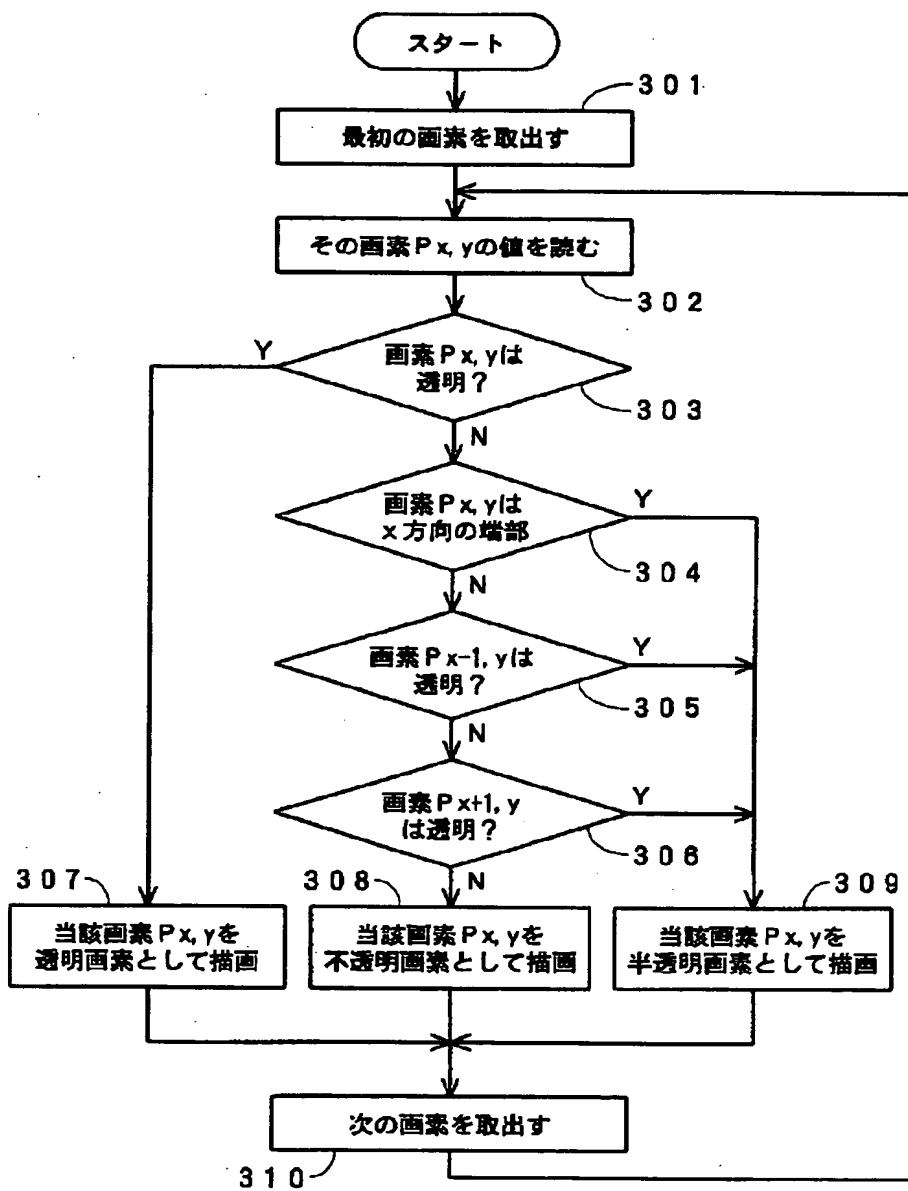
【図18】



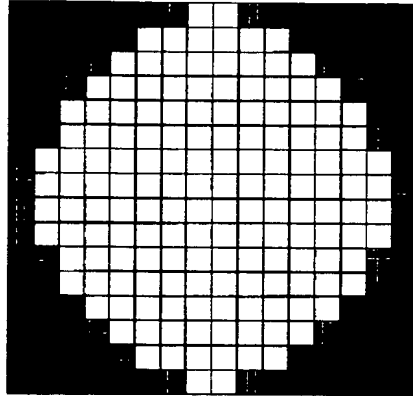
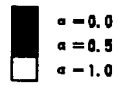
【図19】



【図20】



【図21】



【公報種別】特許法第17条の2の規定による補正の掲載
 【部門区分】第7部門第3区分
 【発行日】平成14年8月30日(2002.8.30)

【公開番号】特開平10-98719
 【公開日】平成10年4月14日(1998.4.14)
 【年通号数】公開特許公報10-988
 【出願番号】特願平8-271909
 【国際特許分類第7版】

H04N 7/24
 G06T 11/00
 H03M 7/30
 H04N 1/41
 5/92

【F1】

H04N 7/13 Z
 H03M 7/30 Z
 H04N 1/41 B
 G06F 15/72 A
 H04N 5/92 H

【手続補正書】
 【提出日】平成14年6月17日(2002.6.17)

【手続補正1】
 【補正対象書類名】明細書
 【補正対象項目名】発明の名称
 【補正方法】変更

【補正内容】
 【発明の名称】 画像データの生成方法、その再生方法
 および画像処理装置並びに記憶媒体

【手続補正2】
 【補正対象書類名】明細書
 【補正対象項目名】特許請求の範囲
 【補正方法】変更
 【補正内容】
 【特許請求の範囲】

【請求項1】 予め定められた所定の大きさの画面領域分の画像データ単位で、画像データを圧縮し、前記画像データ単位の圧縮前の前記画像データのそれぞれの画素について、それが透明であるか不透明であるかを識別する付加データを、前記データ圧縮された画像データ単位毎に対応付けることを特徴とする画像データの生成方法。

【請求項2】 請求項1に記載の画像データの生成方法で生成された画像データの再生方法において、前記データ圧縮された前記画像データを伸長したときに、前記付加データにより透明画素とされている画素は、伸長画像データに関係なく、強制的に透明画素とすることを特徴とする画像データの再生方法。

【請求項3】 予め定められた所定の大きさの画面領域分の画像データ単位で、データ圧縮された画像データと、前記画像データ単位の圧縮前の前記画像データのそれぞれの画素について、それが透明であるか不透明であるかを識別する付加データとが、対とされた入力画像データの画像処理装置であって、前記データ圧縮された単位画像データ毎の前記画像データをデータ伸長する伸長デコード手段と、前記伸長デコード手段からのデータ伸長された単位画像データ毎に、前記付加データにより透明画素とされている画素は、強制的に透明画素とする画像補正手段と、を備えることを特徴とする画像処理装置。

【請求項4】 画素値として透明を表現するための値が割り当てられ画像データにより、透明を含む描画を行う画像処理装置において、前記透明の画素に隣り合う不透明の画素は、半透明にして描画を行うことを特徴とする画像処理装置。

【請求項5】 予め定められた所定の大きさの画面領域分の画像データ単位で、データ圧縮された画像データが記録されると共に、前記画像データ単位の圧縮前の前記画像データのそれぞれの画素について、それが透明であるか不透明であるかを識別する付加データが、前記データ圧縮された画像データ単位との対応関係を持って記録されていることを特徴とする記録媒体。

【手続補正3】
 【補正対象書類名】明細書
 【補正対象項目名】0001
 【補正方法】変更

【補正内容】

【0001】

【発明の属する技術分野】この発明は、例えば、ビデオゲーム機やパーソナルコンピュータのように、高い精度は求められないものの、限られたハードウェア資源の中で、ユーザの入力や演算結果に応じてリアルタイムに3次元物体を表示することが求められるシステムにおける画像データの生成方法、その再生方法および画像処理装置、さらには、画像伝送用の記録媒体に適用して好適なものに関する。

【手続補正4】

【補正対象書類名】明細書

【補正対象項目名】0018

【補正方法】変更

【補正内容】

【0018】

【課題を解決するための手段】上記課題を解決するために、この発明による画像データの生成方法は、予め定められた所定の大きさの画面領域分の画像データ単位で、画像データを圧縮し、前記画像データ単位の圧縮前の前

記画像データのそれぞれの画素について、それが透明であるか不透明であるかを識別する付加データを、前記データ圧縮された画像データ単位毎に対応付けることを特徴とする。

【手続補正5】

【補正対象書類名】明細書

【補正対象項目名】0020

【補正方法】変更

【補正内容】

【0020】この画像データの再生方法によれば、データ圧縮された画像データが伸長復号されたときに、誤差により透明とされるべき画素が不透明となっても、付加データにより、強制的に透明画素に変換される。そして、このように補正された伸長画像データがメモリに蓄えられ、描画装置による描画に用いられる。したがって、付加データをメモリに蓄える必要はなく、付加データの分のメモリ容量は不要であり、しかも、本来の透明部分は透明部分として描画処理されるので、再現画像上においては、非可逆圧縮によるノイズも軽減されることになる。